**High Level Design**

By Jason Barnes

1. **Table**
   1. Column Features
      1. Most column attributes should be editable on the table. Fields such as Column Name will be editable using a normal textbox while other types of attributes may use drop-downs or check boxes.
      2. The user is also allowed to reorder the columns in the table. This should be implemented by allowing the user to drag the columns up and down on the diagram.
2. **Keys**
   1. Primary Keys
      1. Glyph
         1. The Primary Key glyph will be a gold key aligned vertically and placed to the far left edge.
   2. Foreign Keys
      1. Glyph
         1. A Foreign Key will be represented by a silver key pointing to the left.
         2. Groups of keys will be organized by placing a number to the right of the key. For example the diagram below shows two pairs of foreign keys. The pairs are identified by a 1 and 2.
      2. Organization
         1. Multiple columns involved in a relationship should be arranged together programmatically. As shown in the diagram, columns involved in a relationship are vertically aligned. Relationships with the most columns involved will be farther to the left while relationships involving fewer or one column are stacked to the right.
      3. Foreign Relationship Connector
         1. A line connector will represent a foreign key relationship between tables. The connector will directly connect to the columns of each table that are involved in the relationship.
         2. In the event that the relationship involves 2 or more columns from each table then the connect will fork and point directly to the columns. The forked ends will be labeled with a number so that they can be mapped to the correct column. Hovering the mouse over the line connect will show a tooltip that states which table and column is involved in the relationship.
3. **Constraints**
   1. Uniqueness Constraints
      1. Glyph
         1. Some columns may be optional and as a result have the possibility of two options being applied to them.
            1. You can treat the null value as an actual value and apply the uniqueness constraint. This option will be represented by a Solid letter U.
            2. You can ignore the null value and apply the uniqueness constraint to non null values only. This option will be represented by a hollow letter U.

